5

APPARATUS AND METHODS FOR OPTIMIZING TRAFFIC VOLUME IN WIRELESS EMAIL COMMUNICATIONS

10

RELATED APPLICATIONS

This application claims priority to the provisional application entitled "Data Synchronization System Modeling and Optimization for Support of Disconnected Operation and High Data Availability," filed on February 2, 2000, and bearing the

15 serial number 60/179,761.

This application is also related to applications entitled "Apparatus and Methods for Providing Universal Data Synchronization Algorithms by Facilitating Data Synchronization System Design," "Apparatus And Methods For Providing Coordinated And Personalized Application And Data Management For Resource-

20 limited Mobile Devices," and "Apparatus and Methods for Providing Personalized Application Search for Wireless Devices Based on Self User Profiling," bearing serial numbers _____, _____, and _____, respectively. These applications were filed on _____ and all claimed priority to the above provisional application bearing serial number 60/179,761.

25

FIELD OF THE INVENTION

This invention relates to wireless communications. In particular, this invention relates to apparatus and methods for optimizing traffic volume in wireless email communications.

30

35

1

BACKGROUND OF THE INVENTION

Electronic mail (email) has in recent years become one of the most convenient and widely used means of communication. Most existing email systems are based on SMTP for outgoing messages and POP3 or IMAP for incoming messages. In addition, some email systems utilize the so-called extended email protocols that allow electronic deliveries of multi-format data in email messages. These extended email protocols, however, do not address bandwidth utilization or traffic volume optimization of email communications.

Efficient email communications require optimized bandwidth utilization. Bandwidth utilization is especially important in networks having limited bandwidth. Wireless networks, particularly wireless cellular networks, typically have very limited bandwidth than wired networks. Thus, optimizing traffic volume of wireless email communications, thereby improving bandwidth utilization, is especially important.

Thus, it is desirable to provide apparatus and methods for optimizing traffic volume in wireless email communications.

SUMMARY OF THE INVENTION

An exemplary method for optimizing traffic volume in a communications network comprises the steps of receiving a first file, parsing a header portion of the file to find a reference header, extracting an identifier of a second file in the reference header, determining a nearest location to retrieve the second file based on the identifier, and retrieving the second file based on the determining. In one embodiment, the extracting step includes the step of determining a tag associated with the reference header; the tag including the identifier and an address to download the second file. In another embodiment, the step of determining a nearest location includes the steps of examining a local cache for a copy of the second file, examining the reference header for a server address to download the second file if the local cache does not include the second file, and extracting an address of a sender of the first file if the reference header does not include the server address.

Another exemplary method for optimizing traffic volume in a communication network comprises the steps of receiving a first file having a first identifier, generating a tag for the first file based on the first identifier, embedding the tag in a second file, creating an association to the tag in a reference header of the second file, assigning a second identifier to the second file, and sending the second file. In one embodiment,

the exemplary method further comprises the steps of determining a nearest address to download the first file and including the nearest address in the tag.

Yet another exemplary method for optimizing traffic volume in a communication network comprising the steps of receiving a first file, parsing a header portion of the first file for a reference header, extracting an identifier to a second file based on the reference header, examining a local cache for a copy of the second file, and updating the reference header to include an address to the local cache if the copy of the second file is in the local cache. In one embodiment, the extracting step includes the step of determining a tag associated with the reference header; the tag including the identifier.

An exemplary computer program product for optimizing traffic volume in a communications network comprises logic code for receiving a first file, logic code for parsing a header portion of the file to find a reference header, logic code for extracting an identifier of a second file in the reference header, logic code for determining a nearest location to retrieve the second file based on the identifier, and logic code for retrieving the second file based on the determining. In one embodiment, the logic code for extracting includes logic code for determining a tag associated with the reference header; the tag including the identifier and an address to download the second file. In another embodiment, the logic code for determining a nearest location includes logic code for examining a local cache for a copy of the second file, logic code for examining the reference header for a server address to download the second file if the local cache does not include the second file, and logic code for extracting an address of a sender of the first file if the reference header does not include the server address.

Another exemplary computer program product for optimizing traffic volume in a communication network comprises logic code for receiving a first file having a first identifier, logic code for generating a tag for the first file based on the first identifier, logic code for embedding the tag in a second file, logic code for creating an association to the tag in a reference header of the second file, logic code for assigning a second identifier to the second file, and logic code for sending the second file. In one embodiment, the exemplary computer program product further comprises logic code for determining a nearest address to download the first file and logic code for including the nearest address in the tag.

Yet another computer program product for optimizing traffic volume in a communication network comprises logic code for receiving a first file, logic code for parsing a header portion of the first file for a reference header, logic code for extracting an identifier to a second file based on the reference header, logic code for examining a local cache for a copy of the second file, and logic code for updating the reference header to include an address to the local cache if the copy of the second file is in the local cache.

In one embodiment, the logic code for extracting includes logic code for determining a tag associated with the reference header; the tag including the identifier.

BRIEF DESCRIPTION OF THE DRAWINGS

FIGURE 1 schematically illustrates an exemplary prior art network.

FIGURE 2 schematically illustrates an exemplary network in accordance with an embodiment of the invention.

FIGURE 3 schematically illustrates an exemplary client system in accordance with an embodiment of the invention.

FIGURE 4 illustrates an exemplary process in accordance with an embodiment of the invention.

FIGURE 5 schematically illustrates an exemplary server system in accordance with an embodiment of the invention.

FIGURE 6 illustrates another exemplary process in accordance with an embodiment of the invention.

DETAILED DESCRIPTION OF THE INVENTION

Figure 1 schematically illustrates an exemplary prior art email network 100. In the email network 100, a user device A 102 and a user device B 104 are connected to an email server A 106 and an email server B 108, respectively. The user device A 102 and the user device B 104 communicate with each other through their respective servers and via multiple gateways or routers (G/R) 110A-110D across the network 100. For example, a user at user device A 102 may initiate communication by generating and sending an email 1 to a user at the user device B 104. In an exemplary embodiment, the email 1 is first sent to the email server A 106. The email server A 106 sends the email 1 through G/R 110A and G/R 110 B to the email server B 108. The email server B 108 delivers the email 1 to the user at the user device B 104.

Likewise, the user at the user device B 104 may respond to the email 1 by generating and sending an email 2 to the user at the user device A 102. The email 2 is first sent to the email server B 108. In an exemplary embodiment, the email 2 includes a reference to the email 1. Typically, the reference is a copy of the email 1 appended to the email 2. The email server B 108 sends the email 2 through G/R 110D and G/R 110C to the email server A 106. The email server A 106 delivers the email 2 to the user device A 102. Under current standard protocols, a copy of email 1 may be stored temporarily in one or more devices in transit between user A and user B, including in the user device A 102 and/or the email server A 106; thus, the physical copy of the email 1 appended to the email 2 creates unnecessary increased traffic that consumes bandwidth.

Figure 2 schematically illustrates an exemplary email network 200 in accordance with an embodiment of the invention. In the email network 200, the email server A 106 and email server B 108 each has a respective local storage (LS) 204A and 204B for storing email messages managed by each server for a predetermined amount of time. In an exemplary embodiment, the predetermined amount of time is configured in the email servers automatically or manually. Similarly, the user device A 102 and the user device B 104 each has a respective LS 204a and 204b for storing email messages generated or received by each device for a predetermined or an infinite amount of time.

In an exemplary embodiment, the user device A 102 generates an email 1, assigns a unique ID to the email 1, and sends the email 1 to the email server A 106. In an exemplary embodiment, the user device A 102 may save a copy of the email 1 in the local storage 204a temporarily. The email server A 106 sends the email 1 to the email server B 108 via G/R 110. In an exemplary embodiment, the email server A 106 may save a copy of the email 1 in its local storage 204A temporarily. The email server B 108 sends the email 1 to the user device B 104. Again, depending on the current protocol applicable to email server B 108, the email server B 108 may save a copy of email 1 in its local storage 204B temporarily.

When responding to the email 1, the user device B 104 may wish to refer to email 1. If so, in an exemplary embodiment, the user device B 104 creates a tag that refers to the email 1 and an address where a copy of the email 1 can be downloaded. In one embodiment, the tag is created based on the unique ID assigned to the email 1 by the user device A 102. In the response (namely, email 2), the user device B 104 embeds the tag in the email 2, assigns a unique ID to the email 2, and sends the email

2 to the email server B 108. In an exemplary embodiment, a copy of the email 2 may be saved in the local storage 204b of the user device B 104.

After receiving the email 2, the email server B 108 parses the header portion of the email 2 for any reference header that refers to a tag embedded in the email 2. If a reference header is located, the email server B 108 parses the associated tag for a unique ID of the referenced email, in this case the email 1. The email server B 108 examines its local storage 204B for a copy of the referenced mail, or the email 1. If the email server B 108 has a copy of the email 1 in its local storage 204B, it updates the tag to include its address since it is nearer to the user device A 102 (or the destination) than the user device B 104. When the tag has been updated, the email server B 108 may store a copy of the email 2 in its local storage 204B before sending the email 2 to the email server A 106 via G/R 110.

At the email server A 106, the header portion of email 2 is again examined for any reference header. If a reference header is found, the email server A 106 parses the reference header to find a reference to an embedded tag. In an exemplary embodiment, the tag includes an unique ID of the referenced email (i.e., the email 1) and an address (the original address) to download the referenced email. The email server A 106 then checks its local storage 204A for a copy of the referenced email. In this example, if the email server A 106 has a copy of the email 1 in its local storage 204A, the email server A 106 updates the tag to substitute the original address with its own address. In an exemplary embodiment, the original address is substituted only if the email server A 106 is located relatively closer to the user device A 102. Next, the email server A 106 sends the email 2 to the user device A 102. In an exemplary embodiment, the email server A 106 may save a copy of the email 2 in its local storage 204A temporarily before sending it to the user device A 102.

When the user device A 102 receives the email 2, it examines the header portion of the email 2 for any reference header. Upon discovering a reference header, the user device A 102 examines an associated embedded tag in the email 2 for a unique ID associated with a referenced email and an address to download the referenced email. Next, the user device A 102 first examines its own local storage 204a for a copy of the referenced mail. If the user device A 102 does not have the referenced mail in its local storage 204a, it attempts to download the referenced email based on the address in the tag. If no address is included in the tag, the user device A 102 attempts to download the referenced mail from the user device B 104. In this

5

10

15

20

25

30

35

example, if the user device A 102 has a copy of the referenced mail (email 1) in its own local storage 204a, the email 1 is retrieved from the local storage 204a.

If a user at the user device A 102 responds to the email 2 by sending an email 3 that refers to both the email 1 and the email 2, tags for both the email 1 and the email 2 are embedded in the email 3. In an exemplary embodiment, a tag for the email 2 is created based on the unique ID assigned to email 2 by the user device B 104. And the process repeats as described above. As rounds of emails increase between the user device A 102 and the user device B 104, where each new email refers to one or more previously sent email messages, the amount of traffic is substantially reduced by sending embedded tags associated with previous sent email messages instead of the actual email messages.

Figure 3 illustrates an exemplary user device A 102 in accordance with an embodiment of the invention. The user device 102 includes a communications interface 302, a microprocessor 304, a user interface 306, and a memory 308. The memory 308 includes an operating system 310, communications applications 312 (e.g., a browser application), a mail client module 314, a mail encoding module 316, a GUID generator module 318, a tag create and embed module 320, a tag search and referenced mail load module 322, and a local storage 204a. In an exemplary embodiment, the user interface 306 includes a user input device for receiving user inputs and an output display device. In an exemplary embodiment, the mail client module 314 performs basic email functions at the client side and may be a generally available software such as Microsoft Outlook by Microsoft. In one embodiment, the mail encoding module 316 encodes email to ensure each email is understandable and decodable by the recipient.

The GUID generator module 318 generates a unique ID for each new outgoing email. In an exemplary embodiment, the unique ID assigned to each email is a global unique mail ID (GUID). In one embodiment, the GUID is a combination of a four-byte number from the IP address or phone number of the user device A 102 and a four-byte sequence number. The sequence number is unique to each email. For example, if the user device A 102 is assigned an IP address of 137.203.96.28 in a wireless network and the user device A 102 has a current sequence number of 198, the next GUID has 8 bytes representing the numbers: 137, 203, 96, 28, 0, 0, 0, 199.

When creating a new email that refers to a previous email, the tag creation and embedding module 320 creates a tag for the previous email based on that previous

email's GUID. In an exemplary embodiment, the tag is created using the hypertext markup language (HTML) or the dynamic markup language (XML). For example, a tag may be written in the HTML language as follows:

Tag <= [user-readable-description]|

[user-readable-description]|

[user-readable-description]|

[user-readable-description]

In the above example, the <a....> indicates the start of the tag and the indicates the end of the tag. The href is a parameter that defines the referenced email or a section of an email. The mail-ID is to be replaced by the GUID of the referenced email. The start-pos is the start position of the referenced email and should be specified if the referenced position does not begin from the top of the referenced email. In an exemplary embodiment, the start-pos is replaced with an integer number that is equal to the number of bytes from the top of the referenced email. The end-pos indicates the end position of the referenced email and should be specified if the referenced position does not end at the bottom of the referenced email. Like the start-pos, in an exemplary embodiment, the end-pos is replaced with an integer number that is equal to the number of bytes from the top of the referenced email. The user-readable-description is the text which is displayed to the mail recipient when the referenced email cannot be loaded (e.g., when the referenced email is already deleted from all devices in the delivery path).

Next, the tag creation and embedding module 320 embeds the tag in the new email and refers to the tag in a reference header of the new email. When the new email is received, the tag search and reference mail loading module 314 parses the new email's header portion for any reference header that refers to a tag. In an exemplary embodiment, the existing extended mail protocols allow one or more reference headers to be included in a message. In one embodiment, each reference header refers to a tag that specifies a referenced mail and a nearest location to download that referenced email. An exemplary process for loading a referenced mail is described in Figure 4 below.

Figure 4 illustrates an exemplary process for loading a referenced mail. At step 402, an email having at least one reference header is received by the user device. For each reference header, the header is parsed for any reference to an embedded tag (step 404). In an exemplary embodiment, the embedded tag specifies a unique ID of a

010480-0003-999   8   CA1 - 265540.1

referenced email and an address to download the referenced email. Next, whether the referenced email is stored in the local storage is determined (step 406). If the referenced email is stored in the local storage of the user device, load the referenced email (step 408). If the referenced email is not stored in the local storage of the user device (step 406), determine whether a server address to download the referenced email is specified in the tag (step 410). If a server address is specified, load the referenced mail from that server (step 412). Referring back to step 410, if a server address is not specified, load the referenced mail from the sender of the email.

Figure 5 schematically illustrates an exemplary email server A 106 in accordance with an embodiment of the invention. The server 106 includes a communications interface 502, a CPU 504, a user interface 506, and memory 508. The memory 508 includes an operating system 510, communications applications 512, mail delivery applications 514, mail server applications 516, tag search module 518, encoded mail depository 520, and a local storage 204A. In an exemplary embodiment, the communications interface facilitates communications between the server 106 and a network, such as a wireless network. The user interface 506 includes a user input device and an output display device. In an exemplary embodiment, the mail delivery applications 514 facilitates mail delivery to each client. The mail server applications 516 performs mail processing and delivery at the server side. The encoded mail depository 520 stores encoded mail. An exemplary process performed by the email server A 106 is described in Figure 6 below.

Figure 6 illustrates an exemplary process performed by the email server A 106 in accordance with an embodiment of the invention. In particular, Figure 6 illustrates an exemplary process facilitated by the tag search module 518 in the email server A 106. At step 602, an email is received by the server 106. The header portion of the email is parsed by the server 106 (step 604). Whether the header portion contains any reference header is determined (step 606). If not, the process ends (step 608). If the header portion contains at least one reference header, the reference header is parsed for any reference to an embedded tag (step 610). If the reference header refers to an embedded tag, the tag is reviewed for information (step 612). For example, a tag may specify a GUID of a referenced email and an address to download that referenced email. Next, using the information, such as the GUID, the server 106 determines if the referenced email is stored in the encoded mail depository 520 (step 614). If not, the process ends (step 608). If the referenced email is stored in the encoded mail

depository 520, the reference header is updated to include the server's 106 address (step 616). Next, the server 106 determines whether there is any more reference header (step 618). If not, the process ends (step 608). If there is another reference header, the process repeats at step 610.

5

The foregoing examples illustrate certain exemplary embodiments of the invention from which other embodiments, variations, and modifications will be apparent to those skilled in the art. The invention should therefore not be limited to the particular embodiments discussed above, but rather is defined by the claims.

10

15

20

25

30

35